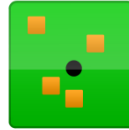


# Location-based messaging on phones

## (via audio networking)

Part II Project by Mark William Hogan



Communication on phones:

- GSM, GPRS, EDGE, UMTS, HSPA, WiMAX, LTE, etc.
- Wi-Fi
- Bluetooth
- NFC (Near Field Communication)

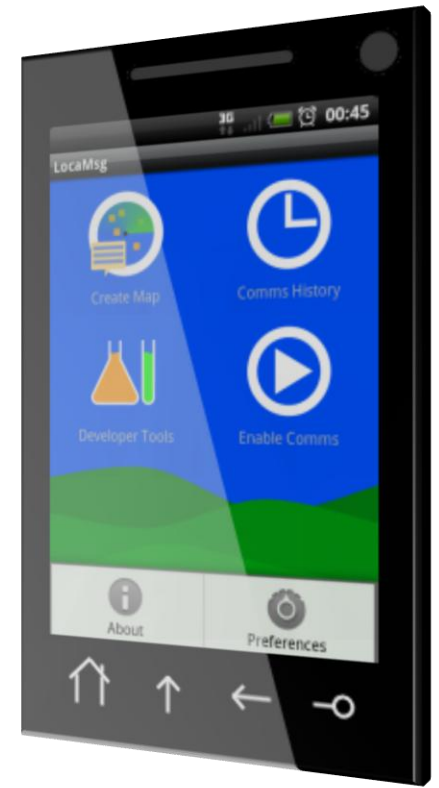
But, by definition, phones have at least one:

- Speaker
- Microphone

Here's an idea:

- Conferences: get a timetable, links to related documents
- Sharing content: send long links
- Exchanging numbers: effortless
- (Restaurants: find out the waiting time, the special of the day)

In the future, resource sharing to save power could make good use of audio networking: low power consumption, security properties.



# What has been done so far?

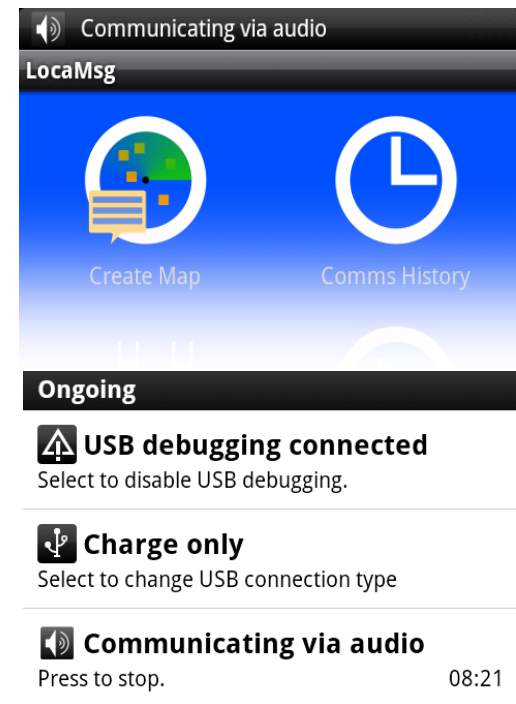
- DTMF-style acoustic modem:
  - Half duplex working reliably using built-in hardware at 20 bps over a few metres
  - Full duplex working less reliably using built-in hardware at 20 bps over a few metres
  - Approx. average CPU usage: < 30% on Qualcomm MSM 7201A and MSM7225, < 10% on Qualcomm Snapdragon QSD8250
- Location of phone: efficient collection of data
- Database: for storing information about discovered phones
- User Interface: including notification for easily instantly stopping communication
- Integration: incoming/outgoing phone call stops audio

## Difficulties

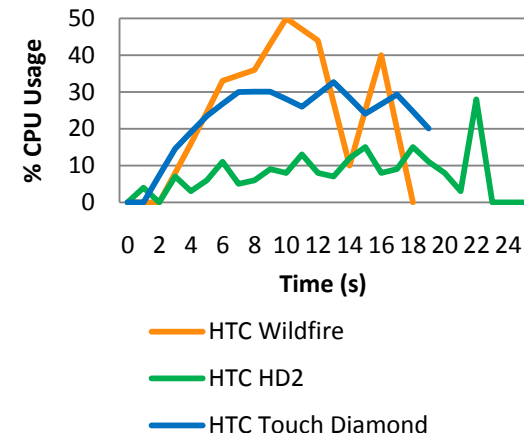
- High audio latency on Android, unreliable APIs
- Real-time processing in resource constrained environment
- Audio hardware variation in power, response curves

## Still in progress

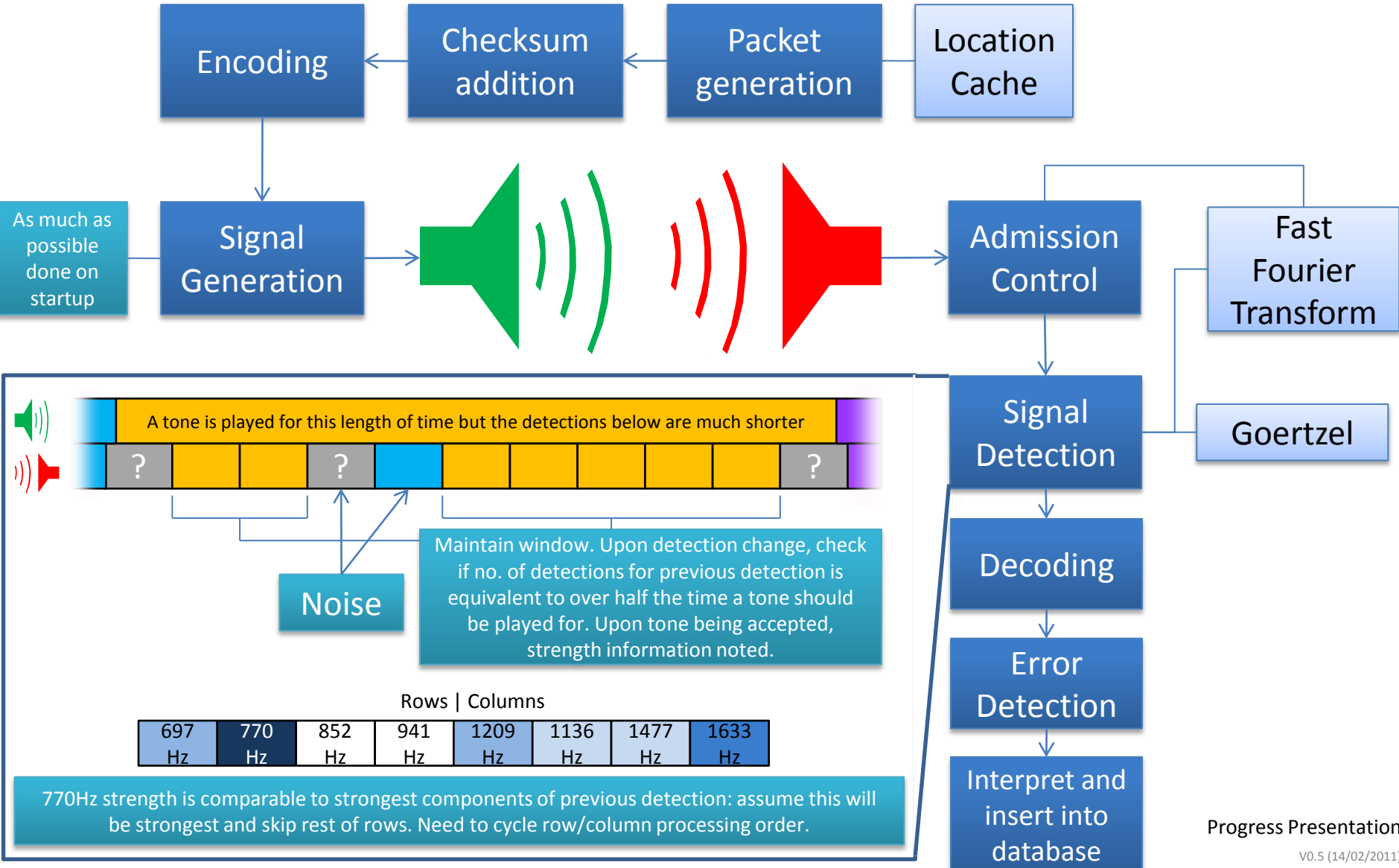
- Integration of media access control and addressing and discovery of multiple devices
- Encoding and packet structure not finalised



CPU Usage during full duplex communication



# Full duplex acoustic modem



# Properties and potential

- Efficiency and optimisation considered at every stage:
  - Raw microphone data can be ignored (i.e. not processed at all)
  - Admission control performs light processing in order to determine if there is significant input
  - Past processing results used to skip some of the signal processing when possible
  - Designed to cope with random noise, loss of signal at any point
  - All non-address tones needed are generated on device start or application start
  - Implementations of signal processing in Java and C: native OpenSL ES API available in Android 2.3 (target is Android 2.2)
- Ultrasound: provides different properties but phone hardware limited to range of a few centimetres
- Security: data is sent unencrypted for performance, malicious equipment to listen in and overcome unreliability of medium will either be noticeable or in close proximity to user
- Works with, rather than against commercial forces: encourages sending of SMS messages